

Willkommen!

Und herzlichen Dank für den Kauf unseres **AZ-Delivery SPI 1,8" TFT-Displays**! Auf den folgenden Seiten gehen wir mit dir gemeinsam die ersten Schritte vom Anschließen des Displays bis zur Anzeige von Schrift und Grafik.

Viel Spaß!



http://flyt.it/TFT_1-8

Das **AZ-Delivery TFT-Display** ist ein **praktisches** und **energiesparsames** Modul, um mehrzeilige Informationen, Grafiken oder einfache Animationen darzustellen, ohne leistungsstarke Rechentechnik einsetzen zu müssen. So können zum Beispiel Sensorwerte von einem Nano V3.0 oder anderem Mikrocontroller ausgelesen und direkt angezeigt werden. Der auf der Rückseite integrierte SD Kartenleser ermöglicht darüber hinaus das Darstellen von Bildern.

Die wichtigsten Informationen in Kürze

» RGB-Display mit 6 bit Farbtiefe (262.144 Farben) » Auflösung: 128 x 160 Px

» Maße (LxBxH)
» Board: 34,5 x 58 x 7,5 mm (inkl. Pins ca. 13 mm hoch)
» Displayeinheit: 34,5 x 46,5 x 2,4 mm
» Anzeigefläche: 32 x 38 mm

» Energieverbrauch
» bis zu 50 mA bei maximaler Hintergrundbeleuchtung

» ST7735 Displaytreiber
» Ansteuerung über SPI

Auf den nächsten Seiten findest du Informationen zur

» **Verbindung mit einem Controller**

und Anleitungen für die

» **Ansteuerung durch zwei Bibliotheken.**

Nützliche Links im Überblick

ST7735 Datenblatt:

» <http://pdf1.alldatasheetde.com/datasheetpdf/view/326213/SITRONIX/ST7735.html>

Programmieroberflächen:

» Arduino IDE:

<https://www.arduino.cc/en/Main/Software> »

Web-Editor: <https://create.arduino.cc/editor>

Arduino-Bibliotheken:

» TFT LCD Library:

<https://www.arduino.cc/en/Reference/TFTLibrary> » Adafruit ST7735 Library:

<https://github.com/adafruit/Adafruit-ST7735-Library>

» Adafruit GFX Library:

<https://github.com/adafruit/Adafruit-GFX-Library>

Interessantes von AZ-Delivery

» AZ-Delivery NEO-6M GPS-Modul:

<https://az-delivery.de/products/neo-6m-gps-module> » Weiteres Mikrocontroller-Zubehör:

<https://www.az-delivery.de/collections/weiteres-zubehoer>

» AZ-Delivery G+Community:

<https://plus.google.com/communities/115110265322509467732> » AZ-Delivery auf Facebook:

<https://www.facebook.com/AZDeliveryShop/>

Verbindung mit einem Controller

Das TFT-Display liegt auf einer Platine, bei der bereits alle notwendigen Ein- und Ausgänge für den Anschluss an einen Mikrocontroller **über eine Pinleiste ausgeführt sind**. Die Anschlüsse für den SD-Kartenleser sind zwar ebenfalls ausgeführt, besitzen aber keine Pins. Du kannst hier selbst entscheiden, ob du Pins oder gleich Verbindungskabel anlöten möchtest.

Um alle Funktionen nutzen zu können, verbinde die Anschlüsse wie folgt mit deinem Controller. Da der Dateneingang sowie der Takt für die Datenübertragung sowohl für das TFT als auch für den Kartenleser gleich sind, empfehlen wir die Verwendung eines Breadboards. Als Controller wird in dieser Anleitung ein **AZ**

Delivery Mikrocontroller eingesetzt.

MC > TFT

3.3 V – LED (Hintergrundbeleuchtung)

13 – SCK (Takt)

11 – SDA (MOSI)

9 – A0

8 – RESET

10 – CS (Chip Select / Adresse)

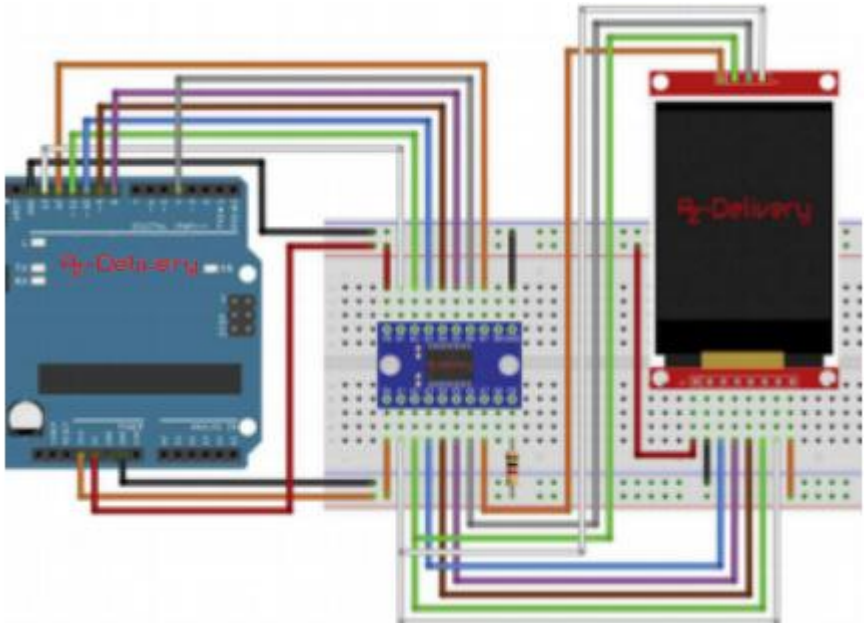
GND – GND

5 V – VCC

MC > SD-Kartenleser

4 – SD_CS (Chip Select / Adresse)

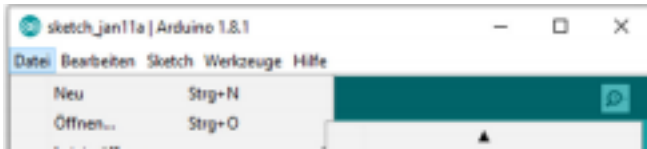
- 11 – SD_MOSI
- 12 – SD_MISO
- 13 – SD_SCK (Takt)



Schrift und Grafiken mit der Standard-Bibliothek

Der im **AZ-Delivery SPI 1,8" TFT** verbaute **ST7735**-Displaytreiber ist so beliebt, dass er über die **TFT-Bibliothek** angesteuert werden kann, welche direkt mit der Arduino-IDE auf deinem Rechner installiert wurde.

Entsprechend kannst du bereits jetzt einen vorgefertigten Sketch auf deinen Controller laden, zum Beispiel "**TFTGraph**":

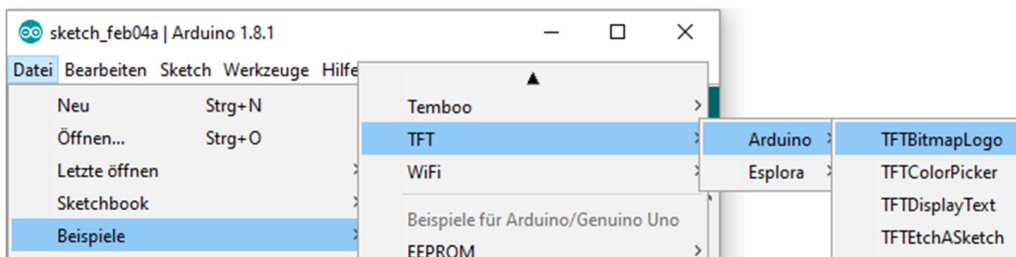


Schaust du dir den Code der Beispielsketch an, wirst du erkennen, dass du mit den folgenden Befehlen bereits einzelne Grafiken und Texte ausgeben kannst:

```
// Hintergrundfarbe (RGB):  
TFTscreen.background(250, 16, 200);  
// Linienfarbe (RGB):  
TFTscreen.stroke(250, 180, 10);  
  
// Linie von Punkt zu Punkt zeichnen  
TFTscreen.line(x1, y1, x2, y2);  
  
// Text schreiben  
TFTscreen.text("Hier steht der Text", x, y);
```

Darstellen eines Bitmap-Bildes

Die TFT-Bibliothek unterstützt nicht nur das Display, sondern auch den integrierten SD-Kartenleser. Um diesen zu testen, öffne den Beispielsketch "TFTBitmapLogo".



Die Verkabelung auf der vorhergehenden Seite ist mit diesem Sketch kompatibel. Was du tun musst, damit er fehlerfrei funktioniert, ist, eine SD-Karte als FAT oder FAT32 zu formatieren und eine bmp-Datei im Querformat mit einer Auflösung von 160 x 128 Px darauf zu speichern. Anschließend benennst du in Zeile 83 des Sketches den Namen des zu ladenden Bildes in deiner gespeicherten Datei um.

```

81  // now that the SD card can be access, try to load the
82  // image file.
83  logo = TFTscreen.loadImage("mein_bild.bmp");
84  if (!logo.isValid()) {
85      Serial.println(F("error while loading arduino.bmp"));
86  }
87  }

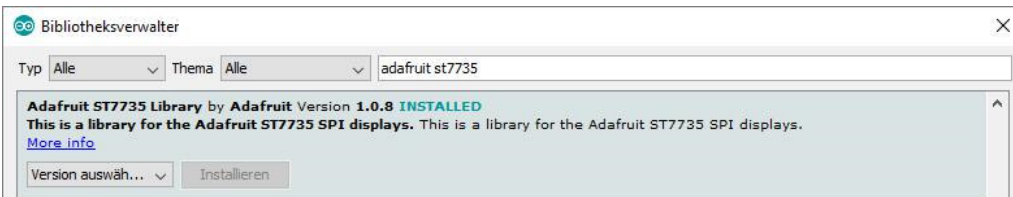
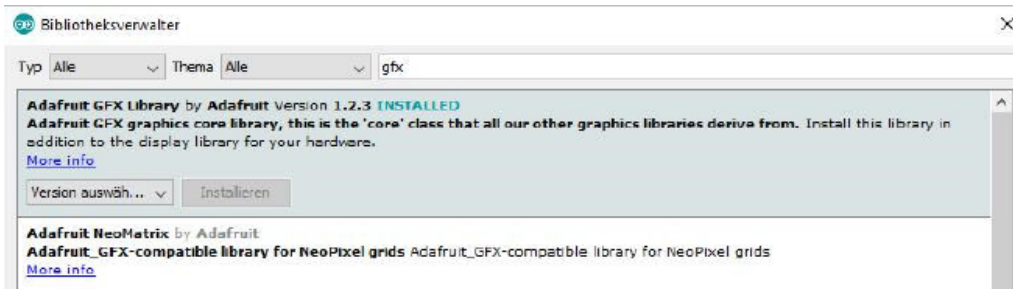
```

Nachdem du den Sketch auf den Mikrocontroller geladen hast, siehst du, wie das Bild auf dem Display aufgebaut wird. Glückwunsch!

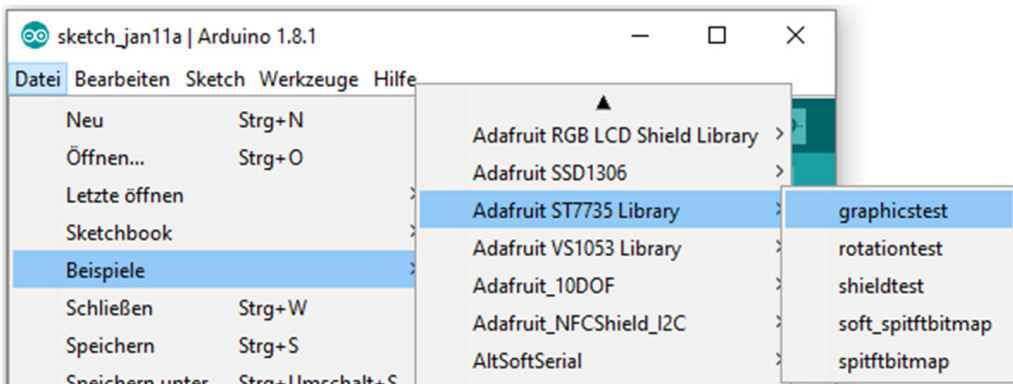
Erweiterte Grafikfunktionen

Wenn du mehr Optionen für die Erstellung deiner Anzeigen haben möchtest, kannst du auf die **GFX-Bibliothek von Adafruit** zurückgreifen. Um sie mit dem Display nutzen zu

können, installiere über den Bibliotheks-Manager die "**Adafruit GFX Library**" und die "**Adafruit ST7735 Library**":



Wenn du nun den Sketch "**graphicstest**" öffnest, findest du jede Menge Gestaltungsmöglichkeiten von der Definition eines einzelnen Pixels über Linien und Rechtecke bis hin zu vorgefertigten komplexeren Formen wie den Medianschaltflächen.



Damit der Code funktioniert, musst du noch eine kleine Änderung vornehmen, da bei Adafruit die **Pins 8 und 9** im Vergleich zur TFT-Bibliothek vertauscht sind. Ändere die

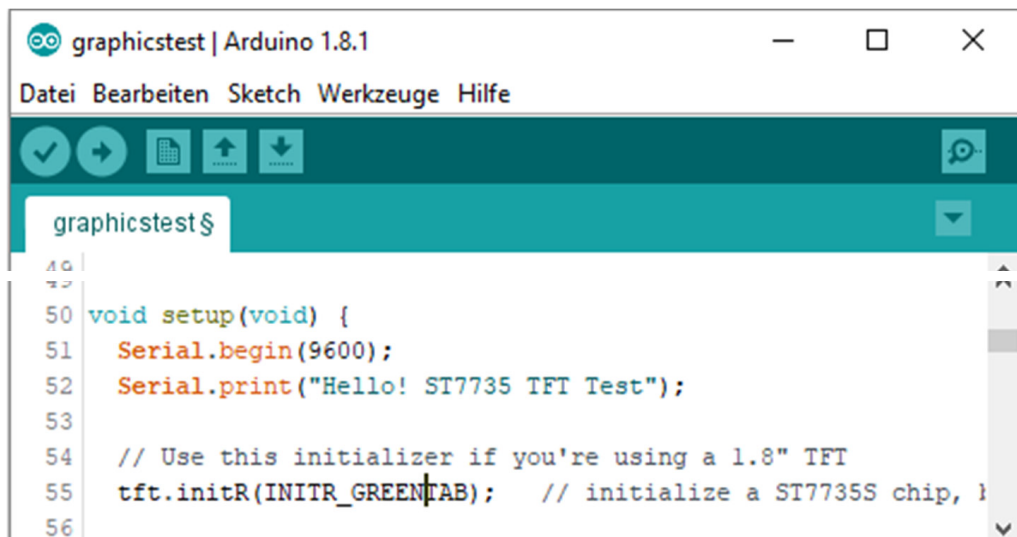
Zeilen 32 und 34 einfach deiner Schaltung entsprechend um.

```
29 // For the breakout, you can use any 2 or 3 pins
30 // These pins will also work for the 1.8" TFT shield
31 #define TFT_CS    10
32 #define TFT_RST    8  // you can also connect this to the Ardu
33                       // in which case, set this #define pin t
34 #define TFT_DC    9
35
```

Nun kannst du auch schon den Sketch hochladen oder du schaust dir den Code an und stellst dir deine eigenen Kreationen zusammen.

Information:

In seltenen Fällen kann es bei dieser Bibliothek vorkommen, dass das Display einen leuchtenden Rand bekommt. Ist dies bei deinem der Fall, ändere in der **setup()**-Methode in Zeile 55 die Variable **"INTR_BLACKTAB"** in **"INTR_GREENTAB"** um.



The screenshot shows the Arduino IDE interface. The title bar reads 'graphicstest | Arduino 1.8.1'. The menu bar includes 'Datei', 'Bearbeiten', 'Sketch', 'Werkzeuge', and 'Hilfe'. The toolbar contains icons for checking, running, uploading, and downloading. The sketch name 'graphicstest' is shown in the top left. The code editor displays the following code:

```
50 void setup(void) {
51   Serial.begin(9600);
52   Serial.print("Hello! ST7735 TFT Test");
53
54   // Use this initializer if you're using a 1.8" TFT
55   tft.initR(INTR_BLACKTAB); // initialize a ST7735S chip, 1
56
```

Du hast es geschafft! Herzlichen Glückwunsch!

Ab jetzt heißt es lernen. Mithilfe des **AZ-Delivery SPI 1,8" TFT Displays** kannst du nicht nur statische Bilder, Grafiken oder Texte visualisieren, sondern zudem sichtbar machen, was dir deine angeschlossenen Sensoren zu sagen haben, ohne dass du auf die Verbindung zu einem Rechner angewiesen bist! Und diese Sensoren sowie weitere Hardware bekommst du natürlich bei deinem Online-Shop auf:

<https://az-delivery.de>

**Viel Spaß!
Impressum**

<https://az-delivery.de/pages/about-us>